Serial No. 09/497,800                          2
Docket No. YOR920000202US1
(YOR.094)

## AMENDMENTS TO THE SPECIFICATION:

**Please amend the specification as follows:**

**At page 1, lines 4-15:**

The present application is related to U.S. Patent Application No. 09/497,802 09/——,——,

filed on February 4, 2000 [[_____]], to Ferrucci et al., entitled "METHOD AND

SYSTEM FOR INTERACTIVE DOCUMENT CONFIGURATION" having IBM Docket No.

YO999-167, U.S. Patent Application No. 09/498,000 09/——,——, filed on February 4, 2000

[[_____]], to Ferrucci et al., entitled "METHOD AND VISUAL APPARATUS FOR

PRESENTING AND NAVIGATING A VARIABLE OBJECT MODEL", having IBM Docket

No. YO999-200, and to U.S. Patent Application No. 09/497,801 09/——,——, filed on February 4,

2000 [[_____]], to Ferrucci et al., entitled "METHOD AND SYSTEM FOR LOOSE

COUPLING OF DOCUMENT AND DOMAIN KNOWLEDGE IN INTERACTIVE

DOCUMENT CONFIGURATION" having IBM Docket No. YO999-201, each in their entirety

incorporated herein by reference.

**At page 3, lines 1-7:**

An interactive document configurator (IDC) assists a user in selecting the appropriate

document components from an archive during a document editing session. The document

components are selected based on a set of configuration rules which may be interactively

updated during a document assembly and editing session. This interactive document

configurator (IDC) is described in the above- mentioned co-pending U.S. Patent Application No.

09/497,802 09/——,——, having IBM Docket No. YO999-167, incorporated herein by reference

in its entirety.

Serial No. 09/497,800                    3
Docket No. YOR920000202US1
(YOR.094)


**At page 3, lines 8-13:**

In a document assembly method using an object model to describe the domain elements that a document's text directly or indirectly refers to, the object model may be captured and linked to the document. This document assembly method and apparatus is described in the above-mentioned U.S. Patent Application No. 09/497,801 ~~09/____,____~~ having IBM Docket No. YO999-201, incorporated herein by reference in its entirety.


**At page 4, lines 11-14:**

For example, assume a company has many stores and many addresses associated therewith including an address (having an address type/format) of a registered office thereof. Such an address (or address type) might be a component which could be in the document or not (e.g., it is optional).


**At page 13, lines 11-19:**

The mapping is determined by a reconciliation algorithm (e.g., described in further detail below). The reconciliation algorithm has mapped variable A from the document component's (e.g., reference numeral 12) to variable 1 in the container assembly 11. The connector 14 holds the mapping of these two variables representing the same domain concept. Thus, the connector is merely an object that stores mappings. Mappings are added to the connector when it is constructed by the reconciliation process, and the connector may be queried for these mappings. Thus, the connector is said to "hold" mappings between variables.

Serial No. 09/497,800                                    4
Docket No. YOR920000202US1
(YOR.094)

**At page 12, lines 18-22:**

Referring to Figure 2, the document component 12 (e.g., source component) is imported into the container assembly 11 composed of previously imported document components ~~17~~. The dotted outline 20 indicates the target location in the container assembly 11 where the document component 12 will be attached after importation.

**At page 13, lines 1-6:**

Shown with each document component is a list of variables referenced within that component. For example, the source component 12 has three variables (e.g., A, B, C in the exemplary embodiment; obviously the source component may have more or less variables as determined by the designer/user). The container assembly 11 has a total of 4 variables (e.g. variables 1-4) referenced in the plurality of document components ~~17~~ held in the container assembly 11.

**At page 13, line 20, to page 14, line 5:**

Additionally, the reconciliation algorithm has mapped <u>component</u> variable B to <u>container</u> variable 3 (of the document component 12) in the container assembly 11 (e.g., containing document). Variable C of document component 12 has no equivalent variable in the container assembly 11 representing a similar domain concept. In this case, a new variable (e.g., variable 5) must be created in the container assembly 11 to represent this domain concept. This addition to the variables in the container assembly is shown in the connector 14 created for this document component importation.

Serial No. 09/497,800                    5
Docket No. YOR920000202US1
(YOR.094)

**At page 16, line 20, to page 17, line 5:**

In one exemplary implementation, the identifying is performed interactively by the user. The system displays ~~by the system displaying~~ component variables and their link expressions next to a representation of elements of the domain model. The user then makes the linking (e.g., associations) by clicking on (actuating via input device) the appropriate variable(s) in the components and matching the variable(s) to an element of the containing document's domain model. Each variable in the component may be matched, but need not be depending upon the user's desires.

**At page 20, lines 20-22:**

Normally, the system is in automatic mode. That is, the automatic reconciler allows automatic reconciliation (linking, matching, identification of associations, etc.) since the components are built from the same model. However, in some scenarios, a manual linking may be desirable.

**At page 21, lines 1-5:**

For example, a manual linking may be desirable ~~Thus, for example,~~ given a plurality of different addresses types (e.g., address with a county designated, address with a signature block designated, address with the state spelled out, address for a mailing label, etc.). However, there might not be any particular person's address (e.g., John Doe's address) present.